

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ  
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

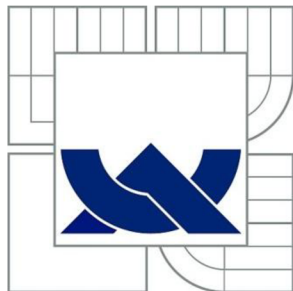
INTELIGENTNÍ MULTIMEDIÁLNÍ PANEL PRO RASPBERRY PI

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

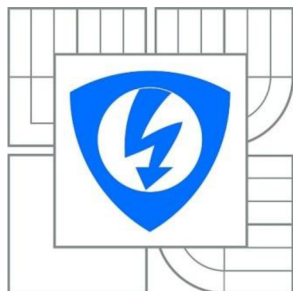
LUKÁŠ BARI

BRNO 2015



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ**

**ÚSTAV TELEKOMUNIKACÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

# **INTELIGENTNÍ MULTIMEDIÁLNÍ PANEL PRO RASPBERRY PI**

A SMART MULTIMEDIA PANEL FOR RASPBERRY PI

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**LUKÁŠ BARI**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. LADISLAV MACHÁŇ**

BRNO 2015



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav telekomunikací

# Bakalářská práce

bakalářský studijní obor  
Teleinformatika

**Student:** Lukáš Bari

**ID:** 154675

**Ročník:** 3

**Akademický rok:** 2014/2015

**NÁZEV TÉMATU:**

## Inteligentní multimediální panel pro Raspberry Pi

### POKYNY PRO VYPRACOVÁNÍ:

Cílem semestrálního projektu je vytvoření konceptu multimediálního panelu pro platformu Raspberry Pi, který umožní přehrávání hudby v různých datových formátech. Multimediální panel musí obsahovat ovládací tlačítka a zobrazovač. Pomocí těchto prvků bude moci uživatel procházet seznamy alb a skladeb, vytvářet fronty skladeb pro postupné přehrávání, apod. Důvodem konstrukce multimediálního panelu je umožnit uživateli použít desku Raspberry Pi pro přehrávání hudby, aniž by musel pracovat v instalovaném operačním systému. Ovládání musí být intuitivní a využívat zavedené způsoby ovládání tak, aby jej mohl využívat kdokoliv.

### DOPORUČENÁ LITERATURA

[1] UPTON, Eben a Gareth HALFACREE. Raspberry Pi: uživatelská příručka. 1. vyd. Brno: Computer Press, 2013, 232 s. ISBN 978-80-251-4116-8.

[2] MANN, Burkhard. C pro mikrokontroléry: ANSI-C, kompilátory C, spojovací programy - linkery, práce s ATMELE AVR a MSC-51, příklady programování v jazyce C, nástroje pro programování, typy a triky ... 1. české vyd. Praha: BEN - technická literatura, 2003, 275 s. ISBN 80-7300-077-6.

[3] DAVIES, J. H. MSP430 Microcontroller Basics. Elsevier Science, 2008. 688 s. ISBN 9780080951577.

**Termín zadání:** 9.2.2015

**Termín odevzdání:** 2.6.2015

**Vedoucí práce:** Ing. Ladislav Macháň

**Konzultanti semestrální práce:**

**doc. Ing. Jiří Mišurec, CSc.**

*Předseda oborové rady*

### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákonníku

## **ABSTRAKT**

Obsahom tejto semestrálnej práce je návrh systému pre multimedialny prehravač založený na platforme Raspberry Pi. Práca obsahuje návrh celého zariadenia ako multimedialneho panelu s možnosťou ovládania priamo, alebo cez vzdialený prístup web rozhrania. Pomocou tohto systému s intuitívnym ovládaním môže užívateľ prechádzať zoznam albumov, skladieb, vytvárať playlisty a podobne.

## **KLÍČOVÁ SLOVA**

Multimedialni panel, Raspberry Pi, hudební přehrávač

## **ABSTRAKT**

The subject of this project is design of a system for a multimedia player based on the Raspberry Pi platform. The work contains a design of the entire device as a multimedia panel with the possibility of direct access or remote access via web interface. With the help of this system with intuitive interface, the user will be able to browse the list of albums, songs, create playlists etc.

## **KEYWORDS**

Multimedia panel, Raspberry Pi, music player

BARI, L. Inteligentní multimedialní panel pro Raspberry Pi. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2015. 46 s. Vedoucí semestrální práce Ing. Ladislav Macháň.

## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma “INTELIGENTNÍ MULTIMEDIÁLNÍ PANEL PRO RASPBERRY PI” jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4  
Trestního zákoníku č. 40/2009 Sb.

Místo .....

.....

(podpis autora)

# OBSAH

ÚVOD .....	8
1 KONCEPT PANELA.....	9
1.1 Logická úroveň.....	10
2 RASPBERRY PI .....	11
2.1 Operačný systém .....	12
2.2 GPIO (General-purpose input/output).....	12
2.3 SPI zbernica (Serial Peripheral Interface).....	13
2.4 MPD .....	13
2.4.1 Klient MPC .....	14
3 ROZŠIRUJÚCE MODULY.....	15
3.1 Mikrokontrolér .....	15
3.2 RTC ( Hodiny reálneho času) .....	16
3.3 Ovládacie prvky .....	20
3.4 Prepínač.....	23
3.5 Vypínač .....	24
3.5.1 Fyzická realizácia vypínača .....	24
3.5.2 Programová podpora vypínača .....	26
4 ZOBRAZOVAČ.....	28
4.1 Programovanie displeja .....	29
5 VZDIALENÝ PRÍSTUP.....	31
5.1 FTP.....	31
5.1.1 Inštalácia a konfigurácia FTP servera a klienta .....	31
5.2 WEB server .....	32
5.2.1 Inštalácia a konfigurácia WEB servera.....	32
5.3 WEB stránka .....	33
5.3.1 Webový MPD klient .....	34

6	REALIZÁCIA .....	35
7	ZÁVER.....	38
8	POUŽITÁ LITERATÚRA.....	39
9	ZOZNAM PRÍLOH .....	42
A	Multimediálny panel .....	43
A.1	Schéma zapojenia.....	43
A.2	Doska plošných spojov .....	44
A.3	Zoznam súčiastok.....	45
B	Obsah CD s prílohou .....	46

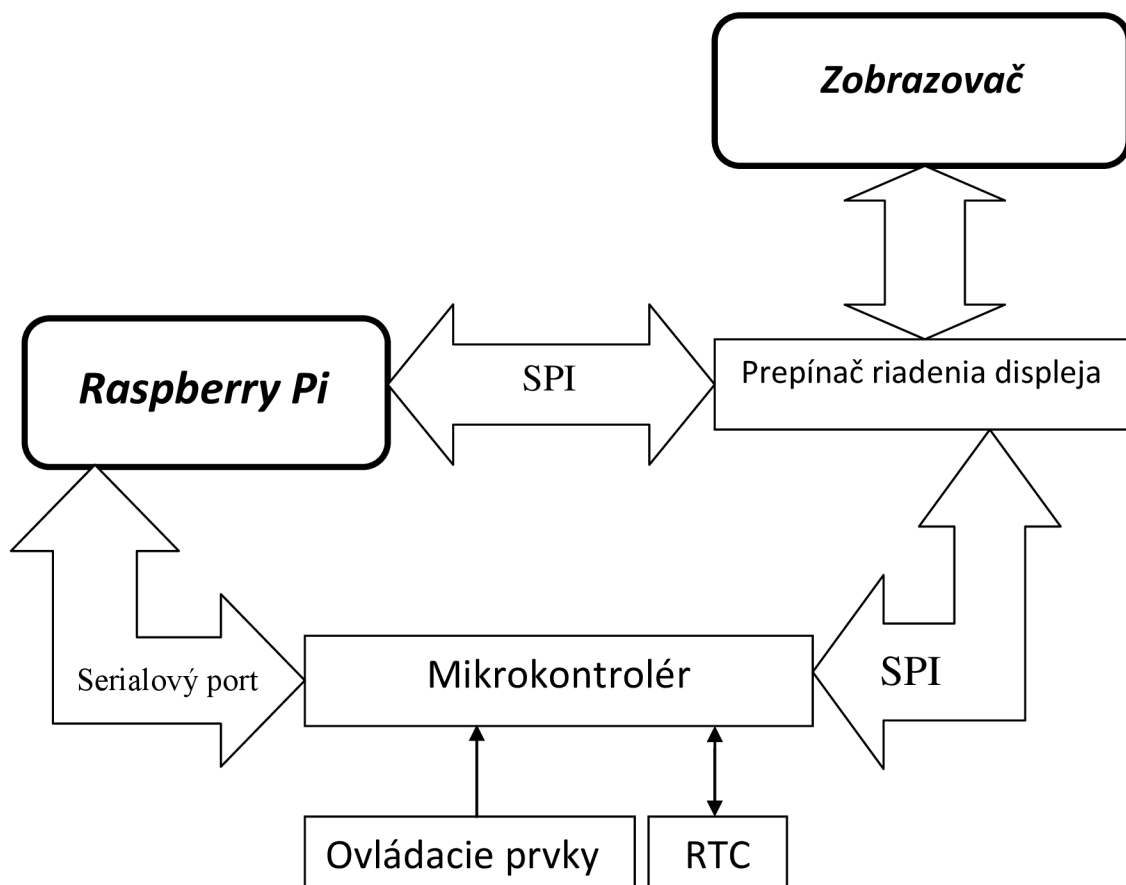
## ÚVOD

Táto práca sa zaoberá návrhom konceptu multimedialného panela s využitím Raspberry Pi. Zariadenie umožňuje prehrávanie a spravovanie hudby s najvyšším pohodlím pre užívateľa. V dnešnej dobe máme doma mnoho rôznych možností prístupu k našej obľúbenej hudbe napr.: domáce kino, stolný PC, rôzne prenosné prehrávače atď. Práca sa zaoberá myšlienkou spojiť čo najviac výhod doterajších zariadení a vytvoriť platformu s hlavným cieľom zabezpečiť pre užívateľa čo najväčší komfort, jednoduchosť a intuitívnosť s použitím známych štandardov pre používanie a ovládanie. Vzdialený prístup cez internet, uložisko dát, praktická veľkosť form factor, intuitívne ovládanie a spravovanie dát, sú základne vlastnosti multimedialného panela. Už dlhšiu dobu je platforma Raspberry – Pi veľmi obľúbená vďaka svojej cene, jednoduchosti a možnosti využitia, preto bola použitá ako mozog celého systému, konkrétne model B+. Ako už bolo spomenuté toto zariadenie je koncipované tak aby mal užívateľ svoje multimédia na jednom zariadení a vedel ich spravovať z akéhokoľvek miesta v domácnosti, s jednoduchým prístupom a pár kliknutiami. Ovládané je vzdialene cez sieť alebo displejom kombinované s tlačidlami na prednej strane. S nimi si užívateľ vyberie svoju skladbu z pomedzi súborov nahranými pomocou FTP. Z toho vyplývajú výhody, že zariadenie nemusíme mať „na dosah ruky“ a bude ho vedieť spravovať a ovládať vzdialené aj viac užívateľov a tak vyberať hudbu a vytvárať playlisty podľa svojej potreby. Zvukový výstup je možný cez stereo konektor 3,5 mm JACK alebo pre viac kanálový zvuk pomocou HDMI.



# 1 KONCEPT PANELA

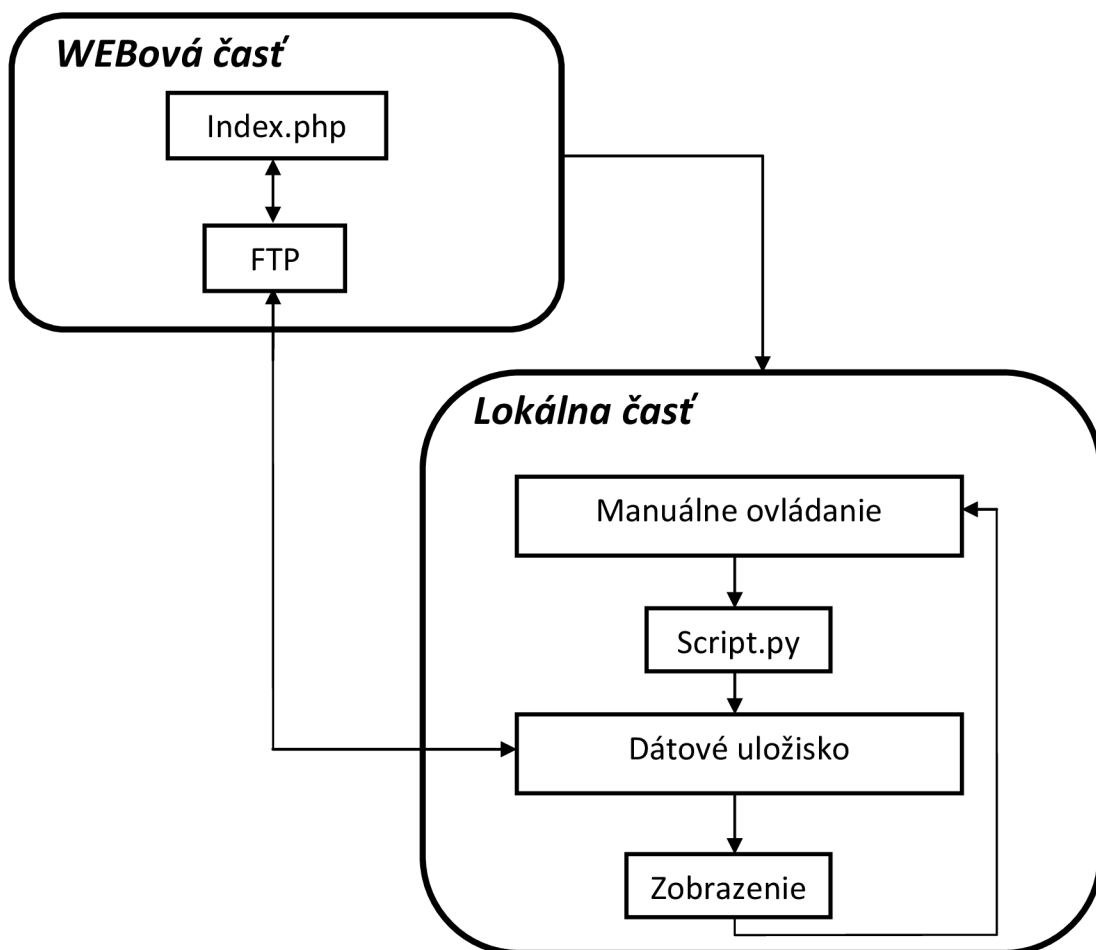
Ako vidieť na obrázku Obr.1 koncepcia sa skladá z troch základných blokov: Raspberry Pi (ďalej už len RPi), zobrazovača, osadeného na doske plošného spoja a rozširujúci modul s integrovanými obvodmi. Hlavná výpočtová a riadiaca jednotka je RPi. Tá však dostáva základné pokyny od mikrokontroléra umiestneného na rozširujúcej doske plošného spoja. Vďaka tomuto integrovanému obvodu v spolupráci s RTC obvodom a elektronickým spínačom je možné zobrazovať stavové informácie ako dátum a čas na displeji bez nutnosti chodu RPi. Ovládanie je realizované dvoma spôsobmi. Prvým sú vstavané hardvérové prvky na pohyb po užívateľskom rozhraní a ovládanie hlasitosti, ktorých funkčnosť zabezpečí spomínaný mikrokontrolér. Druhým spôsobom je vzdialené ovládanie pomocou web stránky.



Obr. 1: Bloková schéma multimediálneho panela

## 1.1 Logická úroveň

Na testovanie rôznych základných súčiastok cez GPIO boli skripty písané pre bash, čo je ideálne len pre jednoduché princípy ovládania. Pre nakonfigurovanie celého systému sa viac hodi komplexnejší skriptovací jazyk. Najviac je použitý Python verzie 2. Najväčšia výhoda sa skrýva vo veľmi jednoduchej syntaxi a v širokej podpore vývojárov, preto je možné ľahko vyhľadať riešenia prípadných problémov. Mikrokontrolér je naopak programovaný v programe C. Keďže mikrokontrolér samotný nemá také výpočetné schopnosti ako procesor použitý v Rpi, použiť tento jazyk bola nutnosť. Princíp činnosti programu na logickej úrovni je vidieť na Obr. 2.



Obr. 2: Princíp činnosti programu na logickej úrovni

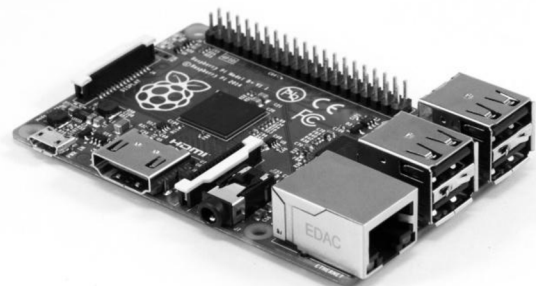
## 2 RASPBERRY PI

Podmienkou zadania bolo použiť ako riadiacu jednotku celého systému platformu RPi, použitá bude verzia B+. Najdôležitejšími novinkami oproti modelu B je viac USB portov, viac GPIO vstupov/výstupov, lepší audio výstup, 4 montážne otvory, lepšie usporiadanie konektorov a nižšia spotreba na 0,5 – 1 W. Základné nezmenené parametre oproti staršej verzii sú tieto:

Tab. 1: Parametre Raspberry Pi B+ [9]

Procesor	ARM 1176JZFS 700 MHz
Grafický procesor	VideoCore IV
Operačná pamäť	512 MB
USB 2.0	4 krát
Obrazový výstup	HD 1080p HDMI; RCA; DSI
Zvukový výstup	4-pólový 3,5 ;HDMI
Pamäťová karta	Slot pre SD kartu
GPIO	40 pinov
Napájanie	Micro USB

Pretože je to plnohodnotný PC má podporu rôznych verzií operačných systémov. Najpreferovanejšie sú hlavne rôzne verzie systému Linux. Aj keď RPi nie je obzvlášť výkonný stroj tak na tento typ projektov je úplne ideálny. Dôležitou výhodou sú hlavne GPIO porty.



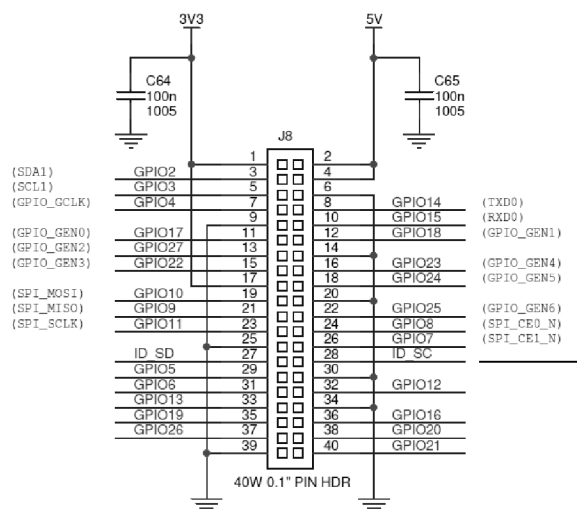
Obr. 3: Raspberry Pi Model B+ [7]

## 2.1 Operačný systém

Samotný výrobca ponúka k počítaču ARMové verzie linuxových distribúcií Debian a Arch a taktiež poskytuje systém na báze Android, napríklad Razdroid pre android 4.0. Pre tento projekt je zvolená oficiálna distribúcia Raspian – Wheezy, ktorá ma základ v Debian 7. Podstata výberu spočíva vo voľbe zlatej strednej cesty medzi rýchlosťou systému a možnosťami, čo nám systém poskytuje. Táto verzia je veľmi obľúbená v komunitou vývojárov na internete a preto má veľkú podporu internetových fór s množstvom materiálov zaoberajúcich sa práve touto problematikou.

## 2.2 GPIO (General-purpose input/output)

Tento konektor umožňuje použiť RPi pre riešenie hardvérových projektov. Vyvedených je 40 GPIO kolíkov. Tieto porty môžu byť vstupné alebo výstupné. Medzi nimi nájdeme komunikačné rozhranie pre I2C a SPI. V tomto projekte, sú používané na komunikáciu so zobrazovačom rozhranie SPI. I2C je použité pre komunikáciu s rozširujúcim modulom RTC. Výstupy napájania 3,3 a 5V sú použité na napájanie zobrazovača a svetelných efektov. Bežná spotreba RPi sa pohybuje okolo 250mA, táto hodnota samozrejme pri záťaži na výstupe rastie. Podľa údajov z katalógového listu reaguje poistka na prekročenie prúdu 2 A na rozdiel od predchádzajúcej verzie kde bola len 1A poistka.



Obr. 4: Raspberry Pi B+ vstup/výstupy [8]

## 2.3 SPI zbernica (Serial Peripheral Interface)

Zbernica slúži k prepojeniu dvoch a viacerých uzlov. Komunikačné uzly sa delia na zariadenia označované ako Master a Slave. Hodinový signal SCLK je vyvedený do všetkých uzlov zbernice a samotná komunikácia prebieha cez vodiče MOSI (výstup zo zariadenia Master) a MISO (výstup zo zariadenia Slave). Nevýhodou tejto zbernice je schopnosť prenosu len na krátku vzdialenosť, kôli nutnosti synchronizácií hodín, a že neobsahuje žiadne potvrdenie doručenia. Tieto nevýhody však v našom riešení nehrajú významu úlohu preto je vhodná na aplikáciu.

## 2.4 MPD

Ďalšou dôležitou súčasťou RPi pre funkcionality celého projektu je prostriedok, ktorý umožní na základe volanej požiadavky užívateľa poskytnúť vlastnosti softvérového hudobného prehrávača. Ideálne pre tieto podmienky je MPD (Music Player Daemon). MPD je veľmi netradičný prehrávač audio súborov. Tradičný v tomto prípade by označovalo softvér, ktorý spustíte, on vám ponúkne svoje ovládacie prvky a pri spustení prehrávania začne dekódovať hudobné súbory a posielat' ich zvukovej karte. Medzi tradičné audio prehrávače môžeme zaradiť napríklad XMMS, MPlayer, Amarok a ďalšie. Už z názvu však vyplýva, že ide o „démona prehrávajúceho hudbu“. MPD funguje úplne inak ako vyššie spomínané programy. Je ukrytý v pamäti a nemá žiadne grafické rozhranie. Namiesto toho má vlastné API, cez ktoré k nemu môžete pripojiť niektorého z veľkej rady klientov. Konfigurácia prebieha v rámci konfiguračného súboru `.mpdconf`.

```
$ cat .mpdconf
port                "6600"
music_directory     "~/mp3"
playlist_directory  "~/mpd/playlists"
log_file            "~/mpd/mpd.log"
error_file          "~/mpd/mpd.error"
state_file          "~/mpd/mpd.state"
```

Obr. 5: Čiastočný výpis súboru `.mpdconf`

Prvý riadok definuje port, na ktorom bude démon počúvať. Ďalej nasleduje dôležitá položka, ktorá určuje adresár, kde máte uloženú svoju zásobu hudby. Nasledujú tri voľby, ktoré nastavujú umiestnenie playlistov, logov a súborov s chybovými hláškami. Posledná položka zapína funkciu, ktorá do zvoleného súboru uloží aktuálne nastavenie vo chvíli, keď démona niečo „zabije“. Nastavenie sa potom znova načíta pri ďalšom spustení. Prehrávanie teda prežije aj reštart počítača a bude pokračovať kde skončilo.

### 2.4.1 Klient MPC

Samotný MPD je k ničomu, pretože ho nemôžeme nijako ovládať. K tomu slúžia spomenutí klienti. Ovládacie príkazy sú jednoduché a intuitívne. V našom prípade ich priamo vpisovať do terminálu nebudeme my ale skript v Pythone ktorý previaže akciu stlačenia ovládacieho tlačidla s funkciou odoslania príkazu pre MPC. Syntax príkazov vyzerá nasledujúco: `mpc [options] <command> [<arguments>]`. Na pozíciu „options“ môžeme napríklad voľbou „-p“ priradiť iný ako základne nastavený port(6600). Ako „command“ teda príkaz, sa používajú najčastejšie:

```
add - na pridanie súboru do playlistu,  
clear - na vymazanie obsahu playlistu  
current - výpis aktuálne hranej skladby  
del - vymazanie skladby z playlistu  
next - nasledujúca skladba  
pause / play - pauznutie / zapnutie prehrávania poslednej hranej skladby  
prev - predchádzajúca skladba  
volume <hodnota> - nastavenie hodnoty hlasitosti (0-100)
```

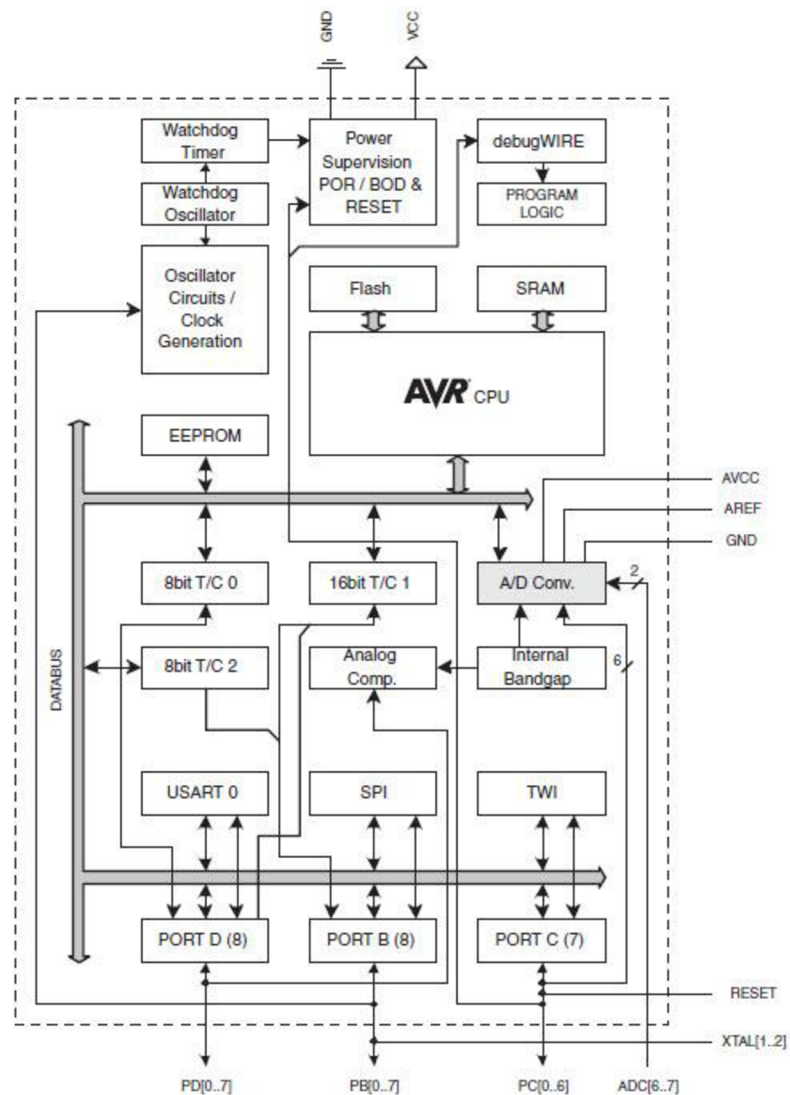
Toto je výpis len najzákladnejších príkazov, existuje ich omnoho viac no tieto sú pre nás podstatné. Ako argument k celému príkazu sa pridáva napríklad konkrétna číselná hodnota (hlasitosť, poradie skladby v playliste...), alebo sa len upresní daný pokyn, len pre konkrétny súbor na ktorý sa bude príkaz vzťahovať.

## 3 ROZŠIRUJÚCE MODULY

Tieto moduly budú realizované ako doplnkové obvody k Rpi, ktoré umožňujú rozšírené možnosti ako hardvérové ovládanie. Taktiež to je spínač pre zapnutie Rpi a celého panela, ako aj spínača, ktorý riadi, kto zo zariadení Rpi alebo ATmega bude mať prístup k zobrazovaču. Z toho vyplýva, že vieme uviesť zariadenie do takzvaného „stand by“ režimu kde zariadenie čaká na zapnutie, no je zároveň schopné poskytovať informácie ako čas, dátum. Je to zaužívaná vymoženosť multimediálnych zariadení no znamená to hlavne úsporu energie, a zamedzenie výpočetnej kontraproduktivity celého systému.

### 3.1 Mikrokontrolér

Mikrokontrolérom sa zariadenie „zobúda“ do chodu a posúva príkazy z ovládania k vykonávaniu procesov pre Raspberry Pi. Umožňuje základné informácie pre užívateľa spolu s obvodom RTC. Hlavnou funkciou je realizovanie ovládania užívateľského rozhrania. To je výhodou hlavne z hľadiska časovania, ktorého presnosť by nebola tak vysoká pri použití priamo procesora RPi. Nemohli by sme ani zaručiť, že vo chvíli výskytu signálu bude mať tento program procesor vôbec pridelený (frekvencia prepínania procesov býva 100 Hz, tento prípad je skôr teóriou). Omnoho lepšie je prenechať obsluhu ovládania mikrokontroléru, ktorý na viac môže zaručiť reakciu v takmer reálnom čase a našemu RPi už len posiela správu o výskyte tejto udalosti. To poslané dáta číta pomocou blokujúceho systémového volania. Výskyt udalosti jednoducho obslúži a potom sa do ďalšieho výskytu zablokuje. Použitý je obvod Atmel ATmega328. Atmel ATmega328 8-bitové AVR® mikrokontroléry sú vysoko výkonné zariadenia RISC, ktoré kombinujú pamäť 32 KB ISP Flash s možnosťou čítania počas zápisu, 1KB EEPROM, 2 KB SRAM, 23 vstupno-výstupných liniek, 32 pracovných registrov, sériový programovateľný USART, a ďalšie. Atmel ATmega328 MCU je schopný spúšťať množstvo inštrukcií v jednom hodinovom cykle, čo umožňuje zariadeniu dosiahnuť priechodnosť blížiacu sa 1 MIPS za MHz pri vyváženej spotrebe energie a rýchlosti spracovania. Je použitý z dôvodu oddelenia vstupných zariadení od výstupných.



Obr. 6: Mikrokontrolér ATmega 328 [12]

### 3.2 RTC ( Hodiny reálneho času)

Hodiny reálneho času (angl. real-time clock, skr. RTC [10]) sú hardvérové alebo menej často softwarové hodiny merajúce fyzikálny (t. j. nie logický) čas. Práve absencia tohto integrovaného obvodu je malou nevýhodou Raspberry Pi. Z toho vyplýva, že bez pripojenia na internet nepozná aktuálny čas a museli by sme ho pri každom spustení zariadenia nastavovať. Je aj možnosť nastaviť interval pri spustení systému, v ktorom si Raspberry Pi porovná hodnotu času. Problém však nastáva ak by bola Raspberry pi veľmi zaťažená, a dochádzalo by k spomaleniu a zníženiu plynulosti. Pri myšlienke tohto projektu sa očakáva a predpokladá prístup na internet aby boli využité všetky možnosti, avšak zariadenie môže pracovať aj v režime offline tzv. bez pripojenia na internet. Preto je implementácia tohto obvodu opodstatnená a taktiež neprináša žiadne



komplikácie z hľadiska implementácie či jej samotnej konfigurácie a je finančne veľmi prístupná. Použitý je RTC obvod DS3231.



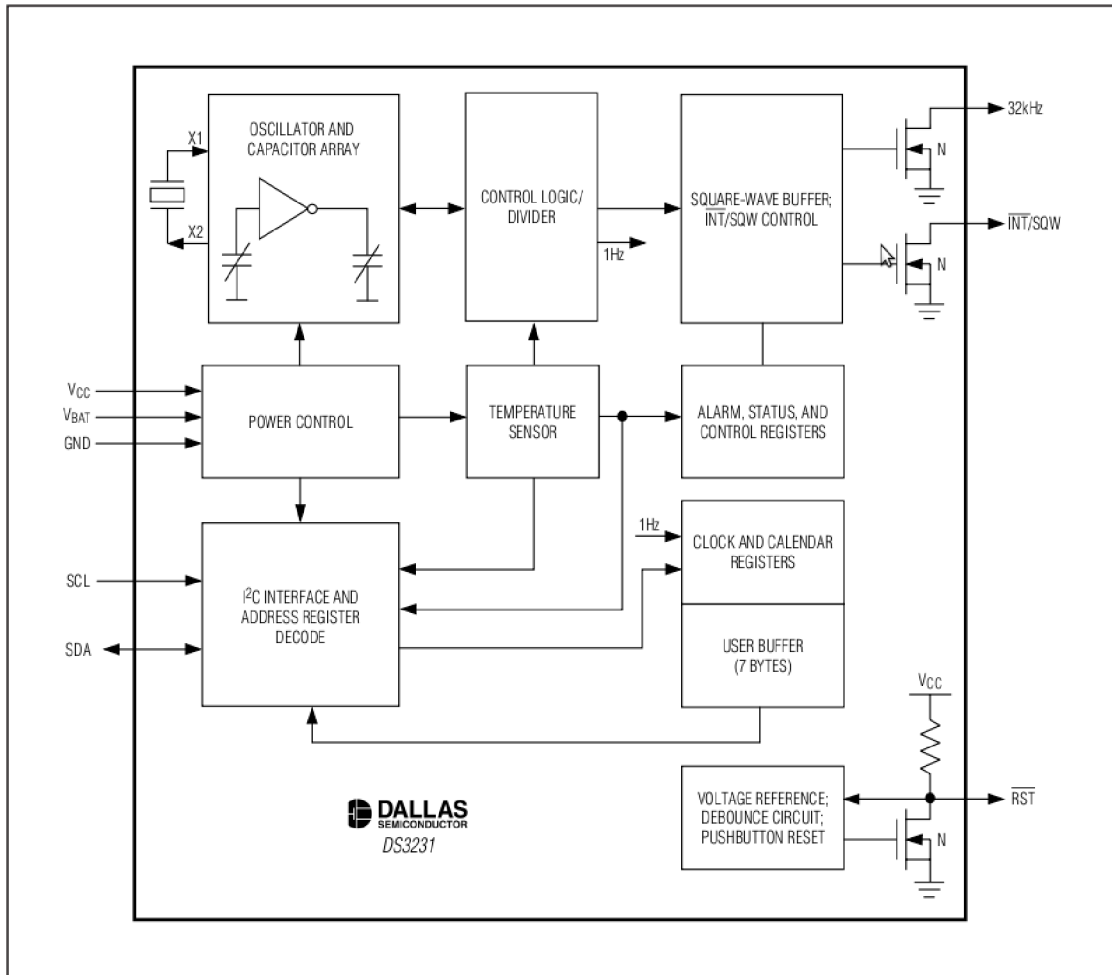
Obr. 7: Integrovaný obvod RTC DS3231 [13]

Je to malý hodinový obvod a je napájaný plochou batériou. Má vysokú presnosť vďaka vnútornému oscilátoru, ktorý neovplyvňujú vonkajšie faktory a preto má odchýlku maximálne pár minút ročne. Obvod reálneho času (RTC) počíta sekundy, minúty, hodiny, dni v týždni, dátum, mesiace, a roky. Dátum je automaticky prispôsobený pre mesiace s menej ako 31 dňami, vrátane korekcie pre prechodné roky. Hodiny pracujú vo formáte 24 hodín alebo 12-tich hodín s indikátorom AM / PM. Dva programovateľné alarmy a programovateľný výstup ponúkajúce režim obdĺžnikového signálu v programovateľných frekvenciách alebo výstup logického stavu. Adresa a dáta sú prenášané pomocou zbernice I<sup>2</sup>C. V prípade tohto multimediálneho panela sa bude tento obvod po nainštalovaní ovládača využívať k zobrazovaniu času na displeji pri vypnutom zariadení. Takže ho bude ovládať mikrokontrolér.

#### **Vlastnosti obvodu:**

- presnosť  $\pm 2$  ppm pri teplotách  $0^{\circ}\text{C}$  až  $+40^{\circ}\text{C}$
- presnosť  $\pm 3.5$  ppm pri teplotách  $-40^{\circ}\text{C}$  až  $+85^{\circ}\text{C}$
- pracovné napätie 3.3 V až 5.5 V
- automatické pripojenie záložnej batérie pre zachovanie behu času
- nízka spotreba obvodu 300 uA pri 5.5 V, v stand-by režime 170 uA
- Obvod reálneho času (RTC) počítajúci sekundy, minúty, hodiny, dni, dátum, mesiace, a roky s prepočtom prechodných rokov a kompenzáciou do roku 2100
- dva programovateľné časové alarmy
- komunikácie pomocou I<sup>2</sup>C zbernice
- integrovaný digitálny teplotný senzor s presnosťou  $\pm 3^{\circ}\text{C}$  v celom rozsahu

- RST výstup / vstup pre externý reset
- RoHS kompatibilita

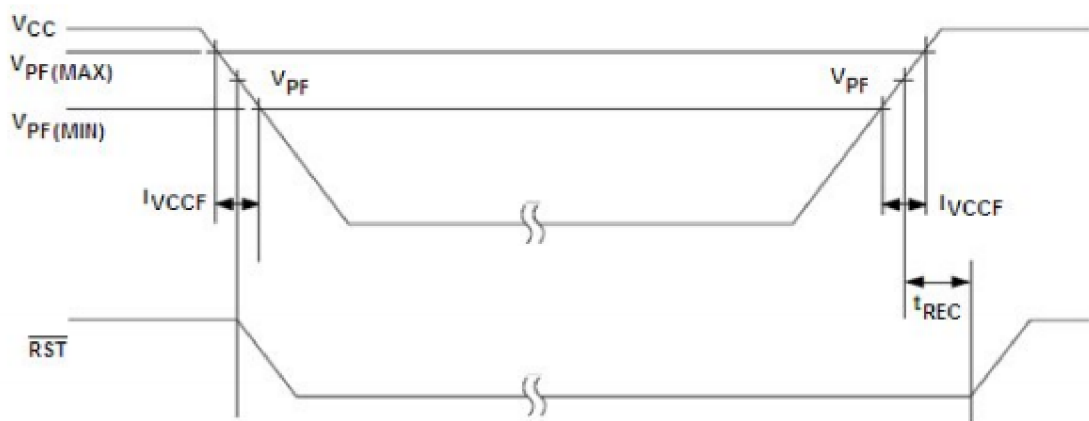


Obr. 8: RTC DS3231 – bloková schéma [27]

Popis jednotlivých blokov :

- **Oscilátor 32kHz:** kontrolná logika obvodu dorovnáva frekvenciu oscilátora pomocou riadených kondenzátorov na základe teploty získanej z interného teplotného senzora
- **Riadenie napájania:** ak je napájacie napätie VCC väčšie ako prahové napätie VPF (min. 2,45 Vmax. 2,70 V) obvod je napájaný z Vcc. Pokiaľ napájacie napätie VCC je nižšia ako VPF a VBAT, obvod DS3231 je pripojený na napájacie napätie batérie VBAT.

- **Snímanie ext. resetu a ovládanie vývodu RST:** DS3231 sníma na pine RST logický stav, pokiaľ prejde na log. 0 dôjde k resetu obvodu a zastavenie behu času len počas trvania log. 0. Ďalšie funkcie vývodu RST je signalizovanie stavu napájacieho napätia. Obrázok 9 znázorňuje priebeh na pinu RST, pri poklese napájacieho napätia na medzu VPF prejde signál RST na log. 0. a následne sa zopne napájanie obvodu zo záložnej batérie pripojené na pin VBAT. Ak napájacie napätie VCC prekročí stanovenú hranicu napájanie z batérie sa odpojí a pin RST prejde do log. 1.



Obr. 9: Priebeh na pine RST [29]

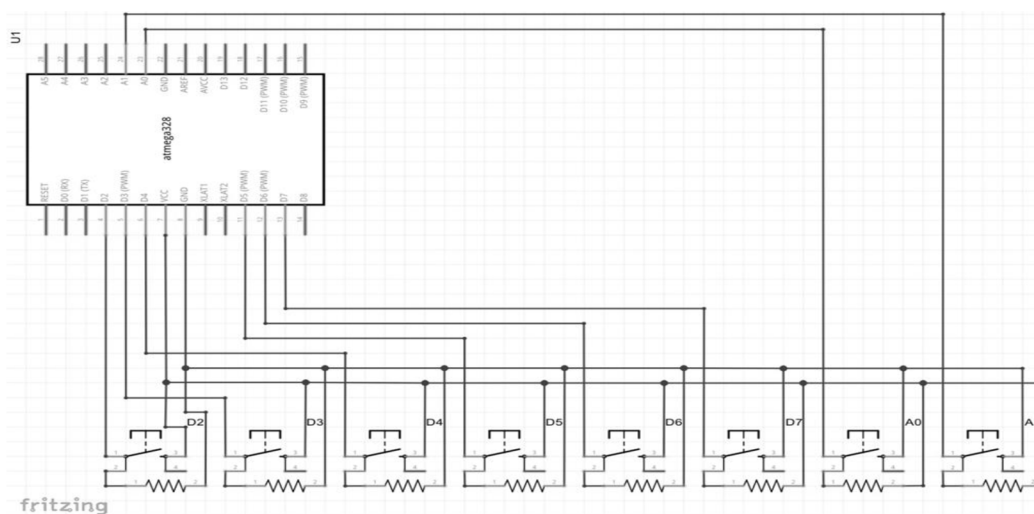
- **Register času a kalendár:** zaznamenáva beh času a umožňuje čítanie sekúnd, minút, hodín, dňa v týždni, dátumu a roku v BCD (binary-coded decimal) formáte pomocou zbernice I<sup>2</sup>C. Voľba medzi 12-tich alebo 24 hodinovým módom sa vykonáva v registri 02h na bitu 6, a to log.1 reprezentuje 12-hodinový mód a log.0 24 hodinový mód. Počítanie dňa v týždni je možné čítať na adrese 03h
- **I<sup>2</sup>C zbernica:** komunikácia obvodu prebieha po zbernici I<sup>2</sup>C prostredníctvom vývodov označených SDA (Data) a SCL (Clock). DS3231 sa na zbernici chová ako Slave, riadiaci obvod označovaný ako Master (riadiaci mikroprocesor generuje obdĺžnikové impulzy na pine SCL o frekvencii 100 kHz

Tab. 2: Popis pinov RTC DS3231 [28]

Pin	Význam	Popis
1	32kHz	32 kHz výstup s otvoreným kolektorom
2	V <sub>cc</sub>	Napájanie 3,3 V až 5,5 V
3	INT/SQW	Výstup pre prerušenie (alarm) / obdĺžnikový výstup nastaviteľnej frekvencie
4	RST	Vstupno/výstupný pin, pre reset obvodu (log. 0) a indikáciu výpadku napájania VCC
5-12	n.c.	Nezapojené, pripájať na GND
13	GND	zem
14	V <sub>BAT</sub>	Vstup pre pripojenie záložnej batérie 2,3 V až 5,5 V
15	SDA	Dátový vstupno/výstupný vývod zbernice I <sup>2</sup> C
16	SCL	Hodinový vstupný vývod zbernice I <sup>2</sup> C, určený pre synchronizáciu prenosu

### 3.3 Ovládacie prvky

Ovládanie po užívateľskom rozhraní pozostáva z tlačidiel pripojených k mikrokontroléru, ktorý prijatý signál spracuje a následne odošle informáciu Rpi, ktoré konkrétne tlačidlo bolo stlačené. Na obr.10 je vidieť zapojenie ovládacích prvkov. D2 až D6 slúžia na ovládanie rozhrania, D7 je tlačidlo napájania, a tlačidlá A0 a A1 sú na ovládanie hlasitosti.



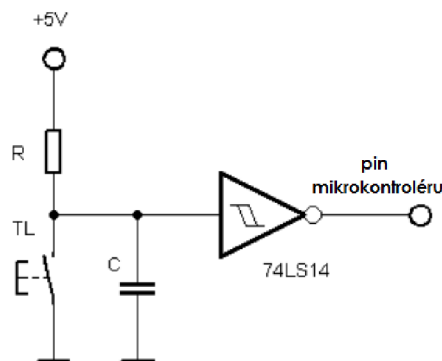
Obr.10: Princíp zapojenie ovládacích prvkov v programe Fritzing

Dôležitým faktorom pri programovaní tlačidiel je ošetriť ich proti zákmitom. To znamená aby na jedno stlačenie bola odoslaná len jedna informácia o jeho stlačení. Bez ošetrenia tohto stavu to nie je vždy zaručené. Na vine je fakt, že sa jedná o mechanický kontakt, ktorý než sa spojí, dôjde k niekoľkým spojeniam a odpojeniam.

## **2 základne typy ošetrenia tohto nežiaduceho javu:**

### **a.) Vhodným fyzickým zapojením tlačidla**

Tu existuje tak isto viacero možností no najbežnejším a najzákladnejším je vytvorenie bezzákmitového tlačidla. To vieme docieľiť pomocou bežného tlačidla v kombinácii s RC článkom (dolný priepust) a Schmittovým klopným obvodom.



Obr. 11: Bezzákmitové tlačidlo [17]

Výpočet hodnôt odporu a kondenzátora závisí na medznom kmitočte dolnej priepusti. Obvykle sa volí hodnota medzi 50 – 100 Hz. Hodnoty je možné určiť vďaka nasledujúceho vzťahu :


$$f_{mez} = \frac{1}{2\pi \cdot R \cdot C}$$

### **b.)Vhodnou úpravou kódu**

Druhou možnosťou ošetrenia zákmitov, je pomocou vhodne napísaného programu. Na detekciu stlačenia je ďalej pridané časové oneskorenie a tiež zistenie z akého logického stavu zmena nastáva. V normálnom stave (tlačidlo nestlačené) je na pine nízka logická úroveň. Pri stlačení dôjde k zmene na vysokú logickú úroveň. Z toho vyplýva, že chceme detekovať prvú zmenu z log. 0 na log. 1 pri stlačení tlačidla. V našom prípade je použitá druhá možnosť, teda ošetrovanie pomocou úpravy kódu. Naša úprava je tá najjednoduchšia a veľmi prostá, ale úplne dostačujúca pre naše potreby.

Docieli sme to „neúplnou“ podmienkou „while“ a to tak že potom ako mikrokontrolér detekuje na tlačidle prvú zmenu z log. 0 na log.1 pošle Rpi informáciu, o ktoré tlačidlo sa jedná, a čaká na argument podmienky „while“, ten však nie je definovaný preto ďalej nevykoná nič a čaká na ďalšiu zmenu log. stavu z 0 na 1. Z toho vyplýva, že ani pri dlhom držaní tlačidla nedôjde k viac ako jednému stlačeniu. Ukážka časti použitého kódu pre nastavenie funkcie tlačidiel a podmienka ošetrujúca zákmity:

```
void loop() {
  for ( int i=0;i<5;i++)
  {
    if(digitalRead(button[i]) == HIGH)
    {
      delay (20);
      if (digitalRead(button[i]) == HIGH)
      {
        Serial.print (button[i]);
        while (digitalRead(button[i]) == HIGH);
      }
    }
    delay (10);
  }
}
```

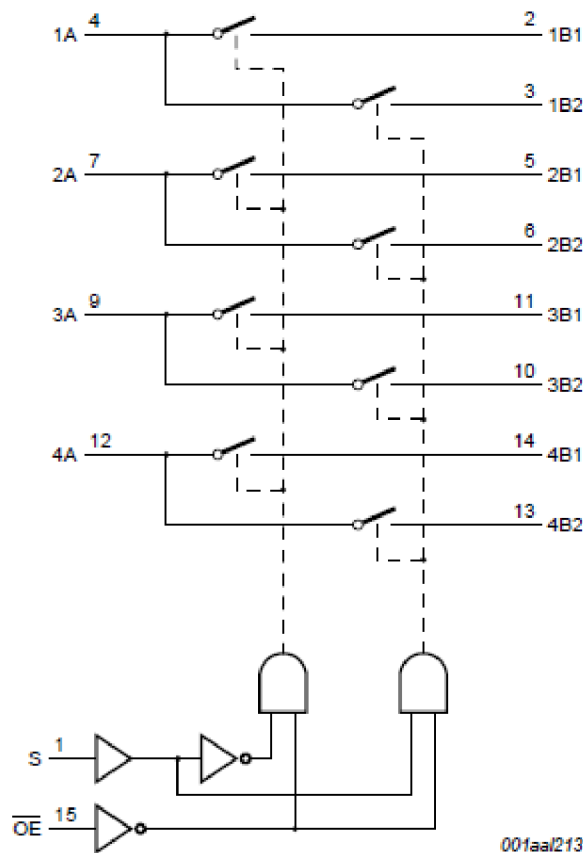


Podmienka ošetrujúca zákmity tlačidla.

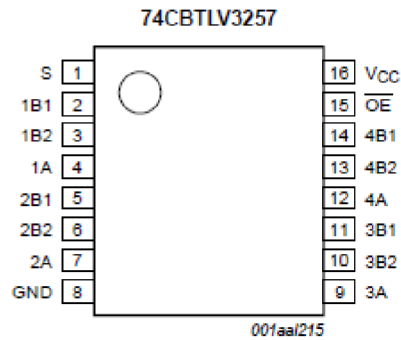
### 3.4 Prepínač

Z blokovej schémy multimediálneho panela [1], vyplýva, že prístup k zobrazovaniu a ovládaniu displeja bude mať tak Rpi ako aj mikrokontrolér. Keďže komunikácia cez zbernicu SPI funguje na princípe master/slave, nebolo by toto zapojenie možné bez ošetrovania už na fyzickej vrstve, kde treba zariadiť aby takzvaný „slave“ (displej) v našej komunikácii mal stále len jedného pripojeného „master-a“ (Rpi/ATmega ). Tu sa ako jediná možnosť ponúka použiť prepínač. Avšak SPI potrebuje na prenos informácií 2 vodiče pre každé zariadenie, prepínanie musí byť riadené elektronicky a rýchlo. Preto je potreba použiť sofistikovanejší prepínací obvod.

Použitý je obvod s označením 74CBTLV3257. Jedná sa o prepínač schopný prepínať s vysokými rýchlosťami. Na strane A indexovanými od 1 po 4 je zapojený spoločný prvok - displej. Na druhej strane prisluchajúcej indexu ľavej strany sú zapojené rovnaké prvky z Rpi respektíve z mikrokontroléru. Spínač S je pin s hodnotou, rozhodujúcou o polohe prepínačov.



Obr. 12: Funkčný diagram 74CBTLV3257 [20]



Obr. 13: Usporiadanie pinov 74CBTLV3257 [20]

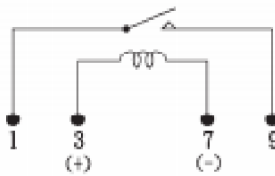
### 3.5 Vypínač

Aj keď je Rpi výborný malý kompaktný linuxový stroj, tak práve na úkor form factoru Rpi stráca na komforte, na ktorý je bežne človek zvyknutý. Ide o to, že Rpi neposkytuje možnosť klasického napájacieho vypínača ako je bežné pri klasických PC, laptopoch a podobných zariadeniach. Napájanie je bežne možné odpojiť len vytiahnutím napájacieho mikroUSB konektora. To nám samozrejme ako riešenie nestačí, preto je potreba pridať obvod riadiaci pripojenie/odpojenie napájania. Nekorektné vypnutie (odpojenie napájania) zariadenia bez zadania príkazu na vypnutie systému, môže poškodiť pamäťové zariadenie, na ktorom je uložený systém ako taký, čo v konečnom dôsledku môže ublížiť Rpi. Z tohto dôvodu je nutnosť ošetriť vypínanie nie len po fyzickej ale aj softvérovej stránke. Priame zadávanie príkazu však v tomto prípade nie je možné, užívateľ nemá k dispozícii bežne prístupný terminál ani grafické rozhranie. Na pozadí všetkého bude opäť bežať skript v Pythone, ktorý to bude všetko riadiť a na reakciu príslušného tlačidla zadá príkaz mikrokontroléru, ktorý vyšle impulz na spínacie relé, ktoré podľa aktuálneho stavu buď zopne ale rozopne spínač s pripojeným napätím.

#### 3.5.1 Fyzická realizácia vypínača

Pre správnu funkčnosť vypínača je nutnosť použiť relé. Konkrétne je použité relé z rady S1A05 od firmy HASCO. Taktiež je potreba použiť externý regulátor napätia. Aj keď je jeden už implementovaný v RPi, tak ho nie je možné použiť a to z dôvodu, že Rpi sa bude vypínať na rozdiel od mikrokontroléra ktorý bude zapnutý vždy a práve on a prepínač 74CBTLV3257 potrebujú 3,3V logiku.

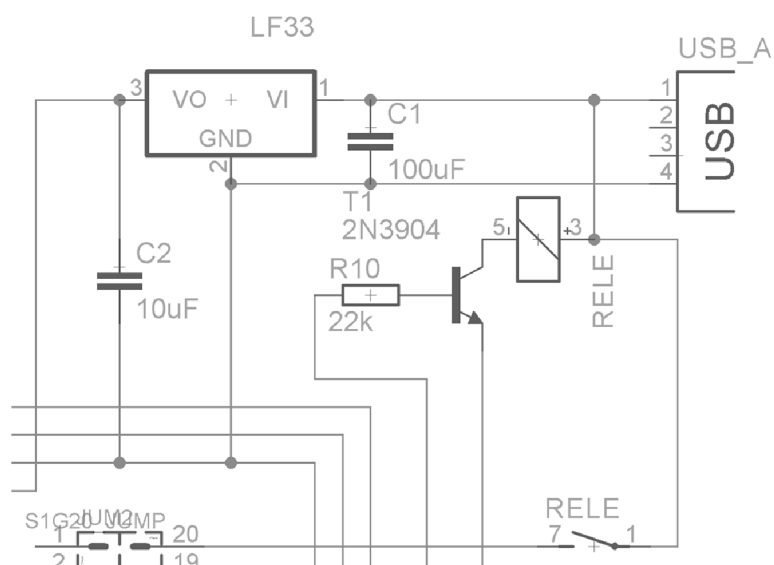




Obr. 14: Schematické zapojenie relé S1A05 [24]

Popis funkcie zapojenia:

- Napájanie bude privedené pomocou USB (5V), je pripojené na regulátor napätia z 5V na 3.3 V, ktoré je použité na napájanie, najmä kôli napät'ovej logike mikrokontroléru a prepínača. Výstup z regulátora je privedený na 3.3V vstup Rpi.
- V ceste medzi konektorom USB a vstupom na regulátor napätia je pripojená cievka spínacieho relé, ktoré ovláda spínač pre zopínanie pripojeného napätia.
- Spínač napájania je ovládaný impulzom z mikrokontroléra, ktorý však treba tranzistorom zosilniť aby mal dostatočnú úroveň na spínanie cievky relé. Použitý je tranzistor 2N3904.
- Taktiež sú pridané kondenzátory na vstup a výstup regulátora. V pomere 10:1.



Obr. 15: Schéma zapojenia vypínača ( v programe Eagle)

### 3.5.2 Programová podpora vypínača

Ako bolo spomenuté pre korektné vypínanie nestačí len odpojiť napájanie ale je potreba nejakým spôsobom informovať Rpi, že k odpojeniu napájania dôjde a ona mala dostatočný čas na to aby sa vypla korektne. Všetko toto bude pokrývať Python, v ktorom máme 2 možnosti ako to spraviť. Pomocou „while“ cyklu alebo pomocou využitia takzvaných „interrupts“ teda prerušení. Použité je druhé riešenie teda báza „prerušení“. Je to najlepší spôsob ako umožniť bezpečné vypnutie pomocou spínača. Zlepšuje účinnosť kódu, ale hlavne minimalizuje zaťaženie CPU v porovnaní s metódou „while“ cyklu. Potom čo sme zabezpečili funkčnosť aj po softvérovej stránke je potreba nastaviť aby sa vyššie spomenutý skript spúšťal vždy už pri zavádzaní systému. To je možné spraviť upravením konfiguračného súboru „rc.local“. Dostaneme sa tam pomocou príkazu :

```
sudo nano /etc/rc.local
```

A pridáme riadok na koniec dokumentu s cestou k nášmu skriptu, napríklad:

```
python /home/pi/start/systemshut.py
```

Po uložení súboru máme plne funkčné ovládanie napájania k multimediálnemu panelu.

#### Zhrnutý proces vypínania/zapínania:

##### Vypínanie :

1. Zariadenie je zapnuté ( tzn. Rpi je zapnuté a ovláda displej , Mikrokontrolér je zapnutý a poskytuje ovládanie)
2. Užívateľ sa rozhodne zariadenie vypnúť, a v ľubovoľnom stave stlačí tlačidlo na vypnutie
3. Tlačidlo pošle impulz do mikrokontroléru, ten vyhodnotí stav tlačidla ako „stlačené“
4. Skript pošle do terminálu Rpi príkaz na vypnutie Rpi: `sudo halt`
5. V Momente stlačenia tlačidla na vypnutie, GPIO 4 na Rpi zmení svoj stav z log1 na log0
6. Zmenou úrovne GPIO 4 prepínač prepína naraz všetky 4 polohy a tým odpája Rpi od riadenia displeja a prevezme ho mikrokontrolér.
7. Mikrokontrolér indikuje stav <HALTING> teda vypínanie. No zároveň už poskytuje hodnoty z RTC (dátum -čas).

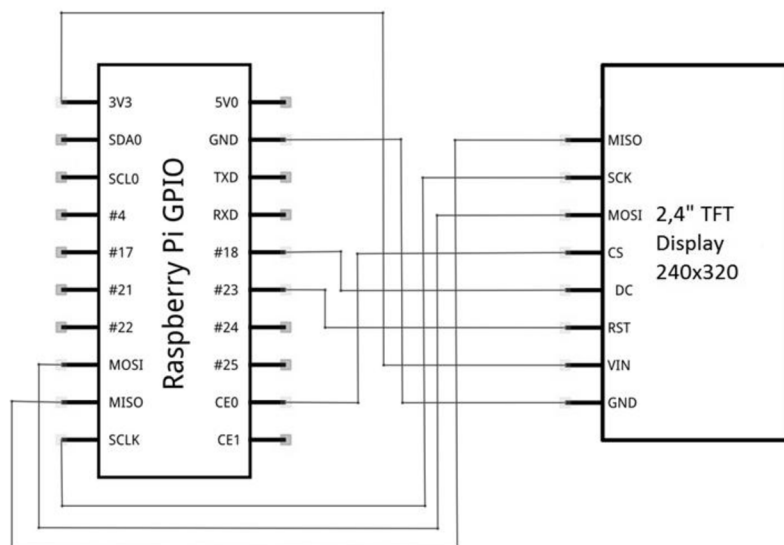
8. Mikrokontrolér posielá tranzistorom zosilnený impulz na relé, ktoré odpojí napájanie Rpi ( tento impulz je však vysielaný s oneskorením 30 sekúnd od stlačenia tlačidla, čo je približný čas korektného vypnutia systému, počas tohto času sa vykonajú body 4 až 7)
9. Uplynie 30 sekúnd, spínač odpojí napájanie od Rpi, a na displeji sa indikuje stav <OFF> spolu s hodnotami z RTC (dátum –čas)

#### Zapínanie:

1. Zariadenie je vypnuté ( tzn. Rpi je vypnuté a odpojené od napájania 5V. Mikrokontrolér je zapnutý, riadi displej a zobrazuje RTC hodnoty)
2. Užívateľ sa rozhodne zariadenie zapnúť a stlačí tlačidlo zapnutia.
3. V tomto momente pošle mikrokontrolér impulz na relé, a zopne napájanie pre Rpi (5V). Rpi sa začne zapínať.
4. Displej stále riadi atmega a indikuje zapínanie zariadenia <BOOTING> opäť súčasne s hodnotami RTC.
5. Počas zapínania Rpi načítava všetky položky zo súboru rc.local, kde sa nachádza aj hlavný skript.
6. Po uplynutí cca 30 sekúnd je Rpi zapnutá → GPIO 4 mení hodnotu z log0 na log1
7. Prepínač S prepína naraz všetky 4 polohy a tým odpája mikrokontrolér od ovládania displeja a prevezme ho Rpi, ktoré ihneď zobrazuje hlavný skript.
8. Zariadenie je zapnuté ( tzn. Rpi je zapnuté a ovláda displej , Mikrokontrolér je zapnutý a poskytuje ovládanie)

## 4 ZOBRAZOVAČ

Zariadenie má k dispozícii na čelnej strane displej uľahčujúci bezprostredné ovládanie. Konkrétne je použitý 2,4 palcový TFT dotykový displej s rozlíšením 240x320 pixelov. Komunikácia je zabezpečená cez SPI zbernicu. Na Obr.16 je vidieť schému zapojenia displeja k Raspberry Pi a zároveň k mikrokontroléru. Displej bude striedavo ovládať Raspberry a mikrokontroler podľa toho v ktorom stave sa bude zariadenie nachádzať. Prepínanie ovládania displeja je zabezpečené pomocou prepínača 74CBTLV3257. Na spomínaný prepínač je pripojený displej pomocou 4 pinov MOSI, SCK, CS, DC. Displej používa ovládač pre TFT displeje ILI9341. Dokáže pracovať v 1,65 V ~ 3,3 V I/O rozhraní. Prekreslovať dokáže frekvenciou 64Mhz. Pretože je knižnica implementovaná v Pythone, nieje potreba použiť kernel modul alebo kernel upravovať. Na zobrazovanie obrázkov, textu a základnej grafiky, ktorá je použitá je to však dostačujúce.



Obr. 16: Schéma priameho zapojenia TFT displeja k Raspberry Pi v programe Fritzing

## 4.1 Programovanie displeja

Použitý je programovací jazyk Python. Najpodstatnejšou časťou zobrazovania na displeji je prepojenie s klientom MPC respektíve MPD. Na začiatok je treba importovať samotného deamona do kódu pomocou `import mpd`. Pred úplným začiatkom samotného zobrazovania treba inicializovať displej `disp.begin()`. Od tohto momentu je displej pripravený dostávať príkazy na zobrazovanie. Pre zaistenie toho, že budeme zobrazovať na čistou plochu, je dobre použiť `disp.clear((0, 0, 0))`. Nie je dobré to ale používať často kôli využívaniu pamäte. Parameter v zátvorke definuje farbu. Súčasťou kódu je ďalej aj otáčanie displeja o 90°. Je to preto, lebo displej je pôvodne orientovaný na portrét, teda na výšku. V takzvanom „idle“ móde teda, v momente keď nie je nič prehrávané sú na displeji zobrazované základne informácie : IP adresa zariadenia, čas - dátum, teplota CPU.

### IP adresa:

Funkciu výpisu ip adresy na displej je zabezpečená krátkym kódom. Pointa je, že sa pokúsi pripojiť na doménu `www.gmail.com`. Adresa ktorá sa o to pokúsi sa vypíše na displej.

```
s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
s.connect(("gmail.com",80))
ipaddress=(s.getsockname()[0])
s.close()
```

### CPU teplota:

```
def getCPUtemperature():
    res = os.popen('vcgencmd measure_temp').readline()
    return(res.replace("temp=", "").replace("'C\n", ""))
```

Samotná výmena informácií prebieha vďaka skúmaniu ID3 tagu MP3 skladby. ID3 tag je dátová štruktúra, súčasť MP3 zvukových súborov identifikujúca samotný MP3 súbor. Ide o metadáta k zvukovým dátam uloženým v tomto formáte, konkrétne sú to: Názov skladby, Interpret, Album, Rok, Žáner atď. Na prehratie MP3 súborov prítomnosť ID3 tagu nie je potrebná.

Tieto metadáta sú posielané a spracovávané MPC klientom. Pre to aby sme odtiaľ izolovali informáciu, ktorú potrebujeme zobrazit' postupne teda umelca, skladbu eventuálne album. Ukážka toho ako získať informáciu z ID3 tagu z mpc.

```
try:
    artist = client.currentsong()['artist']
except:
    artist = "No Artist"

try:
    title = client.currentsong()['title']
except:
    title = "No Title"

try:
    album = client.currentsong()['album']
except:
    album = "No Album"
```

Je dobré využívať takzvanú metódu try/except. Ide o to, že vo vlákne znakov vyhl'adáme položku „artist“ teda umelec a následne to vypíšeme. Ak by bol ID3 tag nekompletný alebo poškodený a nenájde položku „artist“ aby nedošlo k chybe je doplnená podmienka aby vypísal „No Artist“ aby bolo jasné, že vyhl'adanie popisu skladby nebolo úspešné. Obdobne to prebieha aj pre iné parametre skladby.

## 5 VZDIALENÝ PRÍSTUP

Preferovaný prístup k zariadeniu je najmä pripojenie sa vzdialene cez domácu sieť. Cez webové rozhranie pomocou akéhokoľvek zariadenia pripojeného do rovnakej siete ako je aj RPi. Je tu aj ďalšia možnosť komunikácie s panelom a to pomocou FTP servera, ktorý beží na pozadí.

### 5.1 FTP

Tento typ pripojenia je čisto zo správcovskému významu. Poskytuje používateľovi pridávanie respektíve mazanie obsahu v pridelenom priečinku, kde má definované príslušné práva. Nemá mať možnosť priamo spúšťať žiadne médium. Preto komunikácia pomocou tohto protokolu je nevyhnutá len ak sa rozhodne užívateľ upraviť svoju databázu uložených médií. Na komunikáciu sa využívajú dva porty - 20 a 21. Port 20 slúži na prenos dát a port 21 slúži na kontrolu dát a ftp príkazy. Postup nastavenia servera je jednoduchý a prebieha v termináli ako pri klasickom linuxe.

#### 5.1.1 Inštalácia a konfigurácia FTP servera a klienta

Ako prvé treba povoliť na Raspberry Pi SSH pripojenie. To sa po spustení systému spraví následovne. Pomocou príkazu `sudo -i` dostaneme správcovské práva. Ďalej sa príkazom `cd /boot` dostaneme do adresáru `boot` kde príkazom `mv boot_enable_ssh.rc boot.rc` povolíme SSH už pri každom zapnutí zariadenia. Program pre FTP server je použitý `vsftpd`, ktorý je funkčne dostatočne vybavený a má jednoduché nastavenia.

Ako prvé si vytvoríme zložku pre súbory a to vďaka:

```
sudo mkdir /var/ftp
```

Aplikáciu nainštalujeme pomocou:

```
sudo apt-get install vsftpd
```

Po skončení inštalácie sa do konfigurácie dostaneme po napísaní:

```
sudo nano/etc/vsftpd.conf
```

V zobrazenom konfiguračnom liste zmeníme 2 veci. Odstránime znak „#“ (označenie komentára) z riadku `#local_enable=YES` a z riadku `#write_enable=YES`. Na obrázku Obr.8 je zobrazený konfiguračný list programu `vsftpd`. Celý terminál beží cez vzdialený prístup pomocou programu PuTTY. Vďaka tomu je možné konfigurovať Raspberry Pi z domáceho počítača.

```
192.168.1.77 - PuTTY
GNU nano 2.2.4 File: /etc/vsftpd.conf
#
# Run standalone? vsftpd can run either from an inetd or as a standalone
# daemon started from an initscript.
listen=YES
#
# Run standalone with IPv6?
# Like the listen parameter, except vsftpd will listen on an IPv6 socket
# instead of an IPv4 one. This parameter and the listen parameter are mutually
# exclusive.
#listen_ipv6=YES
#
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable=NO
#
# Uncomment this to allow local users to log in.
#local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
#write_enable=YES
[ Wrote 144 lines ]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^H To Spell
```

Obr.17: Konfiguračný súbor programu vsftpd

Tým je konfigurácia FTP servera a užívateľa hotová. Ako ftp klienta môžeme použiť napríklad programy FileZilla alebo TotalComander. Tam už jednoducho a intuitívne do pripraveného adresára vložíme nami vybrané súbory.

## 5.2 WEB server

Vzdialená užívateľská prístupová metóda k multimedialnému panela je pomocou web servera, fungujúcom na pozadí. Používateľ si jednoducho v rámci domácej siete cez internetový prehliadač otvorí stránku, kde má k dispozícii plný obsah úložného priestoru ako z FTP tak aj z rozširujúcich pamäti, ktoré môžeme pripojiť hardvérovo.

### 5.2.1 Inštalácia a konfigurácia WEB servera

Ako základ WEB servera je požitý apache2. Pre jeho inštaláciu napíšeme do terminálu:

```
sudo apt-get install apache2 php5 libapache2-mod-php5
```

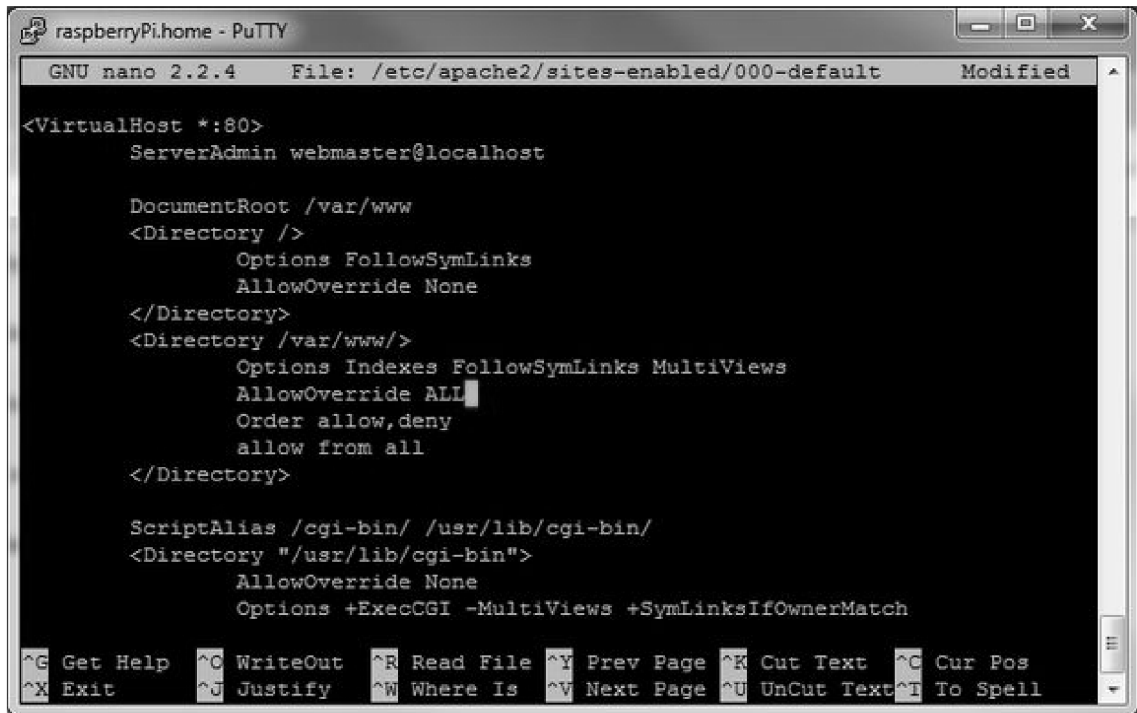
Potom prichádza na rad samotná konfigurácia. Opäť je treba modifikovať konfiguračný súbor ako pri zavádzaní FTP v kapitole 2.1.1.

Pre zobrazenie konfiguračného listu je potreba zadať do príkazového riadku:



```
sudo nano /etc/apache2/sites-enabled/000-default
```

Možnosť prepisovania súborov upraviť zmenou nastavenia „*AllowOverride None*“ na „*AllowOverride ALL*“ viz. Obr.18.



```
GNU nano 2.2.4 File: /etc/apache2/sites-enabled/000-default Modified
<VirtualHost *:80>
  ServerAdmin webmaster@localhost

  DocumentRoot /var/www
  <Directory />
    Options FollowSymLinks
    AllowOverride None
  </Directory>
  <Directory /var/www/>
    Options Indexes FollowSymLinks MultiViews
    AllowOverride ALL
    Order allow,deny
    allow from all
  </Directory>

  ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
  <Directory "/usr/lib/cgi-bin">
    AllowOverride None
    Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
  </Directory>

^G Get Help ^C WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^I To Spell
```

Obr.18: Konfiguračný súbor programu Apache2

### 5.3 WEB stránka

Na prístup k panelu je možné použiť akýkoľvek zariadenie s možnosťou pripojenia na domácu sieť. Keďže prehliadanie stránok na mobilnom zariadení a klasickým PC prináša isté problémy týkajúce sa rozlíšenia, bola implementovaná do kódu www stránky takzvaná jQuery knižnica. Taktiež implementáciou tejto knižnice redukuje počet riadkov kódu. jQuery je ľahká cross-browser JavaScript knižnica, ktorá kladie dôraz na interakciu medzi JavaScriptom a HTML. Vydal ju v roku 2006 vývojár John Resig. Je používaná na viac ako 34% z 1,000,000 najnavštevovanejších web stránok, je aktuálne najpoužívanejšia knižnica. Je slobodný a otvorený software pod MIT licenciou. jQuery syntax je navrhnutá pre jednoduchšiu navigáciu dokumentu, výber DOM elementov, vytváranie animácií, spracovanie udalostí, a vývoj Ajax aplikácií. jQuery tiež poskytuje možnosti pre vývojárov na vytváranie pluginov postavených na tejto JavaScript knižnici. Táto knižnica zvyčajne

existuje ako jeden JavaScript súbor, obsahujúci všetky funkcie pre DOM, udalosti, efekty, a Ajax. Práve posledne spomenutý Ajax bol taktiež použitý pre plynulý chod stránky bez nutnosti znovu načítavania pri pohybe po stránke. Výsledkom je, že stránka pôsobí dojmom aplikácie a užívateľ nelistuje medzi kopou podstránok. To urýchluje a spríjemňuje používanie stránky. Do webovej stránky je jQuery vložená pomocou odkazu na lokálnu kópiu alebo jedným z mnoho kópií dostupných z verejných serverov.

Príklad vloženia :

```
<script type="text/javascript" src="jquery.js"></script>
```

Najobľúbenejší a základný spôsob zavedenia jQuery funkcie je použiť `.ready()` funkciu.

Príklad :

```
$(document).ready(function() {  
    ... //kód  
});
```

### 5.3.1 Webový MPD klient

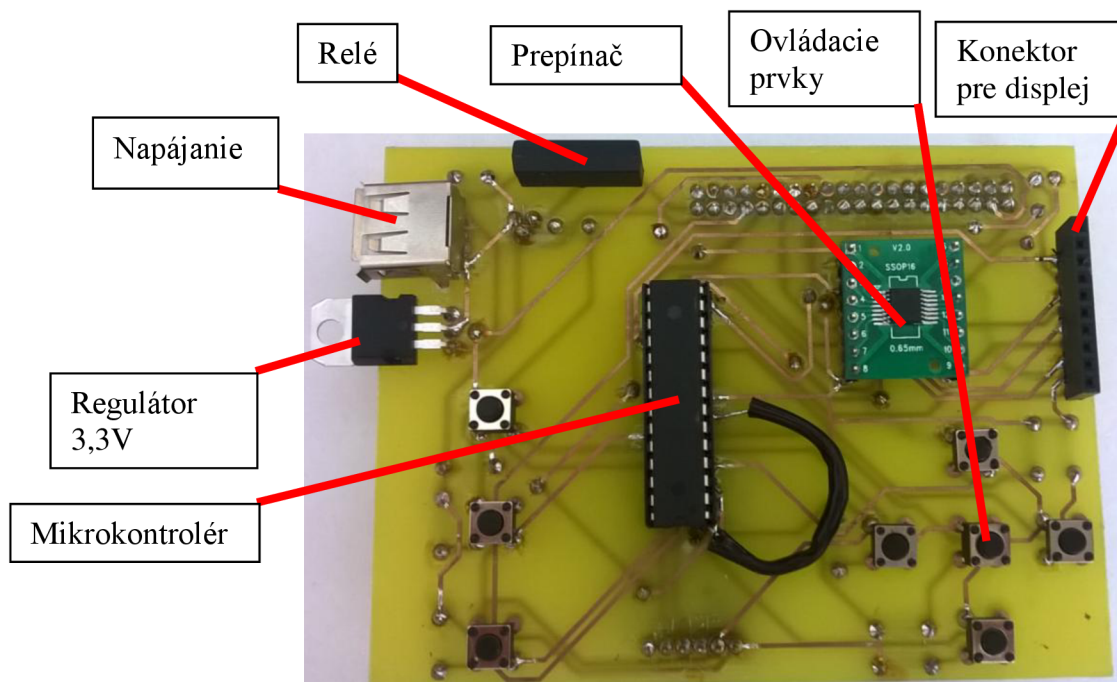
Opäť je potreba pre podporu ovládania pomocou web stránky použiť „nástroj“ schopný komunikovať s našim MPD daemonom, ktorému preposiela príkazy a informácie, ktoré MPD spracuje a vďaka skripte v pythone preposiela stavy prehrávania na displej. Použitý je RelaxXPlayer. Tento klient je postavený práve na spomínanom javascripte a AJAXe. Pre jeho použitie bolo potreba zaviesť respektíve nakonfigurovať skôr spomínaný php server. Na obrázku 19 je vidieť zobrazenie stránky pre ovládanie MPD. Vpravo dole sa nachádza časť kde sú umiestnené všetky súbory z FTP servera. Výberom sa presunú do poľa nad ním kde sa nachádzajú len vybrané skladby a tvoria takzvaný „playing queue“, respektíve frontu prehrávania.



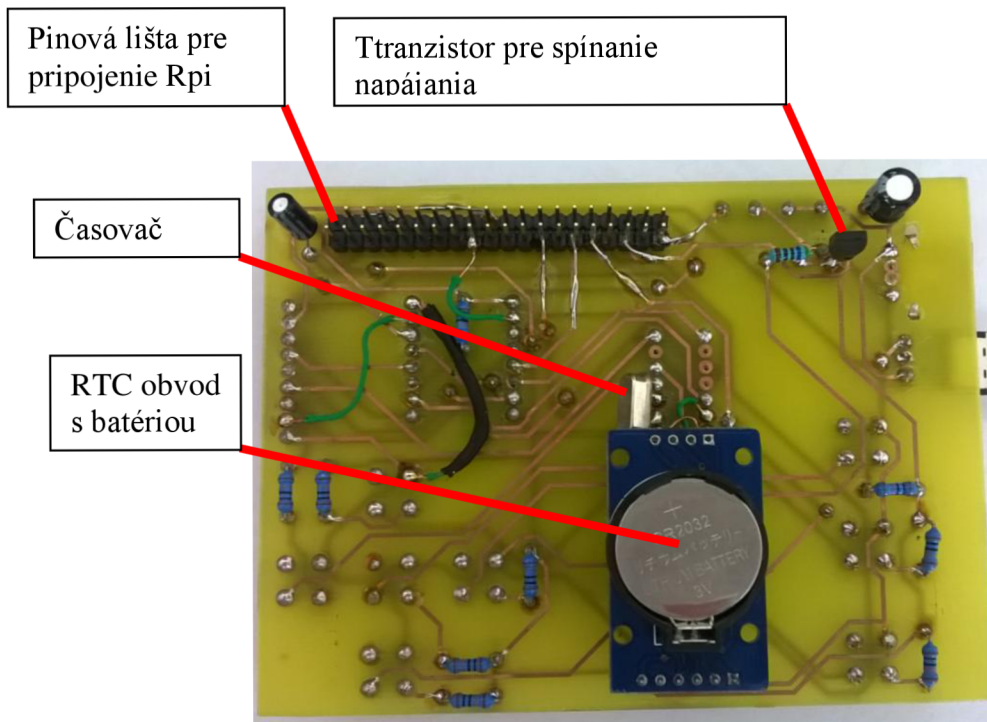
Obr.19: Ukážka webovej stránky pre vzdialené ovládanie

## 6 REALIZÁCIA

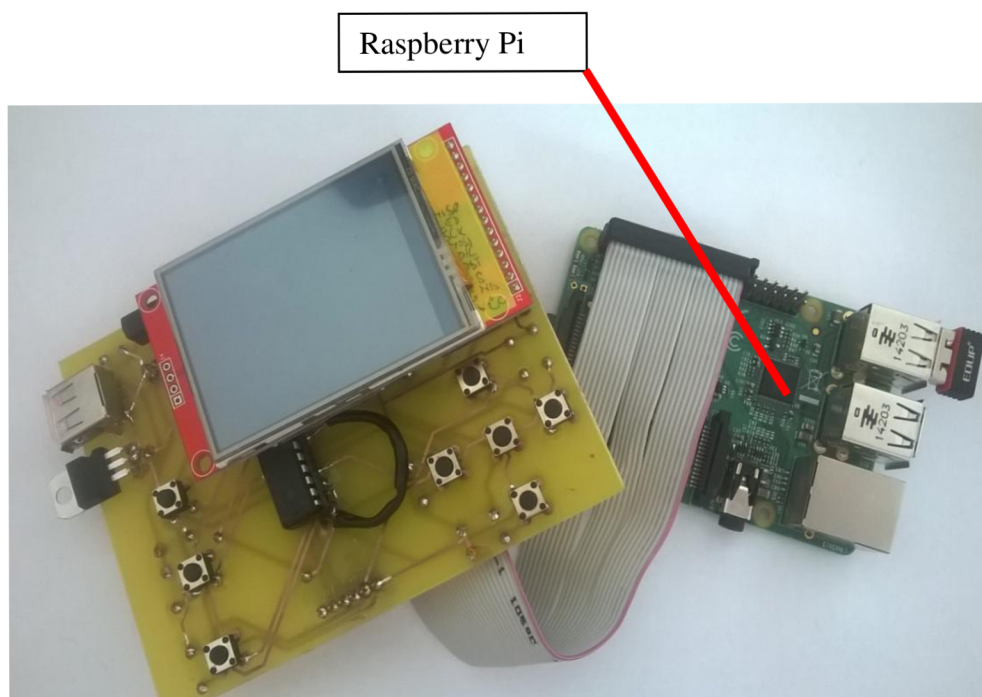
Pri praktickej realizácii je návrh dosky plošného spoja určený pre dvojsvrtvé dosky z dôvodu udržania kompaktnej veľkosti. Osadenie súčiastok je vidieť na obrázkoch 20 až 22.



Obr.20: Osadenie súčiastok – vrchná strana

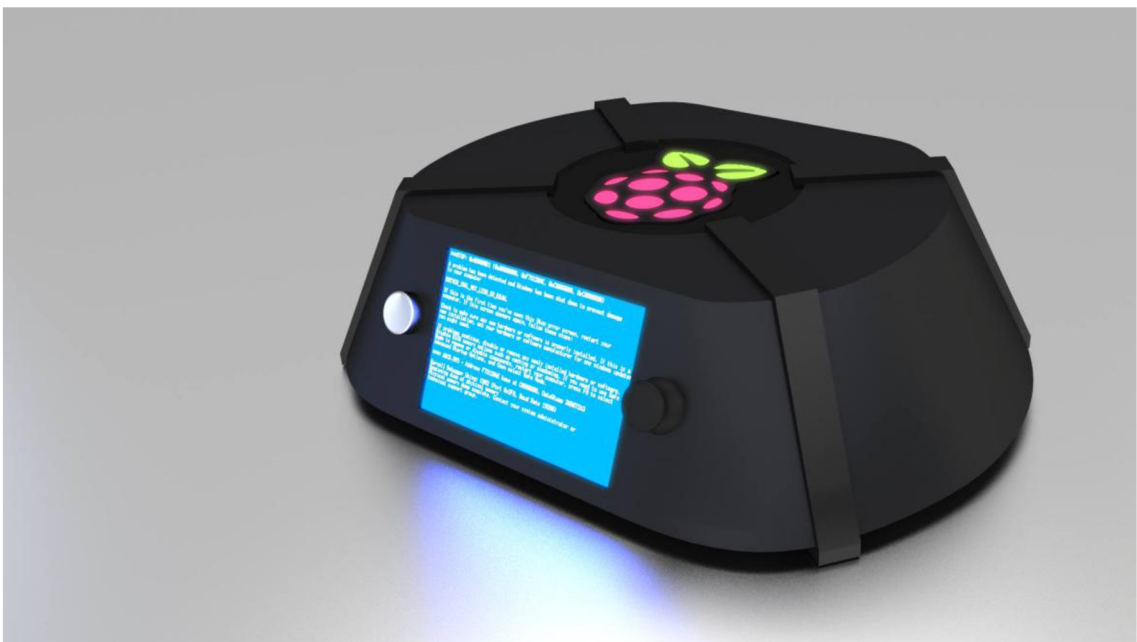


Obr.21: Osadenie súčiastok – spodná strana



Obr.22: Výsledná DPS s pripojeným Raspberry Pi

Do budúca táto práca môže slúžiť ako podklad k finálnemu produktu. Nasledujúca časť nie je súčasťou praktickej časti ani výstupom z práce, ide čisto len o predstavu výsledného produktu. Pri návrhu bolo potrebné rozmýšľať a zohľadniť základné prvky pre praktickosť. Zariadenie bude zložené z jednotlivých komponent poskladaných dokopy. Okrem Raspberry bude puzdro obsahovať jednu dosku plošného spoja s osadenými obvody a súčiastkami. Z obr.18 vidieť, že displej bude osadený pod malým uhlom zhruba 15° z dôvodu, že sa nepredpokladá umiestenie zariadenia v úrovni očí a týmto vychýlením zlepšime výsledné pozorovacie uhly pre užívateľa. Taktiež zošikmenie čelnej časti zlepši ovládanie a to v zmysle, že ovládací prvok je 5smerový joystick to znamená, že na potvrdzovanie bude treba zatlačiť a mohlo by to spôsobovať neustály posun zariadenia. Z toho dôvodu budú pridané gumené prvky na podstavu aby zamedzili pohybu. Výsledný 3D model bol navrhnutý a vypracovaný v softwarovom programe Blender, konkrétne verzia 2.6.



*Obr.18: Prvotný 3D dizajn multimedialneho panela*

## 7 ZÁVER

Cieľom tejto bakalárskej práce bolo navrhnuť systém multimediálneho panela pre domácnosť s možnosťou kombinovaného ovládania. Súčasťou je aj na základe týchto požiadaviek celú myšlienku realizovať prakticky ako hotový produkt. Obsahom bolo predovšetkým, zoznámiť sa s možnosťami Raspberry Pi, základnou správou linuxového systému, konfigurácia web a ftp servera a jednoduché skriptovacie metódy v jazyku Python. Podstata zariadenia je v možnosti ovládania na diaľku pomocou www stránky, no kombinuje túto možnosť aj s hardvérovým ovládaním a zobrazovaním na displeji. Výhodou zariadenia je kompletná správa hudobných súborov a to aj viacerých užívateľov v jednom a zároveň v kompaktnom form-factore. Ďalšou výhodou je možnosť neustálej kontroly a prístupu k zariadeniu, keďže vďaka nízkej spotrebe mikrokontroléra, môže zostávať zariadenie v stand-by režime akokoľvek dlho a nie je nutnosť zariadenie úplne vypínať neustále po ukončení prehrávania. Hlavným prínosom tejto práce, sú predovšetkým znalosti pri riešení tejto problematiky, keďže obsahuje samotný návrh funkcií, softvérovú aj hardvérovú stránku panela ako aj konečný dizajn zariadenia. Výsledné zariadenie pozostáva z Raspberry Pi a rozširujúcich modulov na doske plošného spoja. Zariadenie plní všetky žiadané funkcie na základe zadania práce.

## 8 POUŽITÁ LITERATÚRA

[1] UPTON, Eben a Gareth HALFACREE. Raspberry Pi: uživatelská příručka. 1. vyd. Brno: Computer Press, 2013, 232 s. ISBN 978-80-251-4116-8.

[2] MANN, Burkhard. C pro mikrokontroléry: ANSI-C, kompilátory C, spojovací programy - linkery, práce s ATMEL AVR a MSC-51, příklady programování v jazyce C, nástroje pro programování, tipy a triky ... 1. české vyd. Praha: BEN - technická literatura, 2003, 275 s. ISBN 80-7300-077-6.

[3] DAVIES, J. H. MSP430 Microcontroller Basics. Elsevier Science, 2008. 688 s. ISBN 9780080951577.

[4] TAFT, Darryl K.. jQuery Eases JavaScript, AJAX Development [online]. eWeek : 2006-08-30, [cit. 10. 12. 2014]

[5] Pokusy s GPIO konektorem [online]. 2013, poslední aktualizace 4. 8. 2013 [cit. 15. 12. 2014]. Dostupné z URL: <<http://www.astromik.org/raspi/07.htm>>.

[6] GME produkty [online]. [cit. 2014-12-10]. ISBN DS1302 Trickle-Charge Timekeeping Chip. Dostupné z: <https://www.gme.sk/img/cache/doc/433/112/ds1302-datasheet-1.pdf>

[7] Raspberry Pi Model B+. [online]. [cit. 2014-12-12]. Dostupné z:[http://cdn.shopify.com/s/files/1/0188/6794/products/B\\_Pi\\_1\\_of\\_4\\_1024x1024.JPG?v=1405095882](http://cdn.shopify.com/s/files/1/0188/6794/products/B_Pi_1_of_4_1024x1024.JPG?v=1405095882)

[8] Raspberry Pi model B+ GPIO. [online]. [cit. 2014-12-16]. Dostupné z: <http://www.raspberrypi.org/wp-content/uploads/2014/04/bplus-gpio.png>

[9] Datasheet Raspberry Pi Model B+. [online]. [cit. 2014-12-12]. Dostupné z: <https://www.adafruit.com/datasheets/pi-specs.pdf>

- [10] COOPER, Tyler. What is an RTC?. In: [online]. [cit. 2014-12-12]. Dostupné z: <https://learn.adafruit.com/ds1307-real-time-clock-breakout-board-kit/what-is-an-rtc>
- [11] Český manuál k relačnímu databázovému systému MySQL. Popis nejužitečnějších funkcí, včetně příkladů. [online]. [cit. 2014-12-12]. Dostupné z: <http://www.junext.net/mysql/>
- [12] Atmega - bloková schéma [online]. [cit. 2015-06-01]. Dostupné z: <http://www.mouser.com/images/microsites/atmega328block.jpg>
- [13] Connduino-Real time clock (RTC) module [online]. [cit. 2015-05-19]. Dostupné z: <http://connduino.com/articles/admin/connduino-intrefaces-add-modules#sthash.p2y0L8WN.dpbs>
- [14] Linux man page. [online]. [cit. 2015-04-09]. Dostupné z: <http://linux.die.net/man/1/mpc>
- [15] Display font. [online]. [cit. 2015-04-09]. Dostupné z: <http://www.dafont.com/>
- [16] JQuery Fundamentals. [online]. [cit. 2015-04-09]. Dostupné z: <http://jqfundamentals.com/>
- [17] Základy ovládání mikrokontroléru ATmega. [online]. [cit. 2015-04-10]. Dostupné z: <http://www.tajned.cz/2013/10/osetreni-zakmitu-na-vstupnim-pinu-mikrokontroleru/>
- [18] MomentaryPushButton. [online]. [cit. 2015-04-10]. Dostupné z: <http://arduinotutorial.altervista.org/wp-content/uploads/2011/03/MomentaryPushButton-01a-L.jpg>
- [19] Tactile Push Button Switch Momentary Tact 6x6x6mm. [online]. [cit. 2015-04-10]. Dostupné z: [http://www.aliexpress.com/store/product/6x6xH6mm-Tactile-Push-Button-Switch-Momentary-Tact-1000pcs/107624\\_511657037.html](http://www.aliexpress.com/store/product/6x6xH6mm-Tactile-Push-Button-Switch-Momentary-Tact-1000pcs/107624_511657037.html)



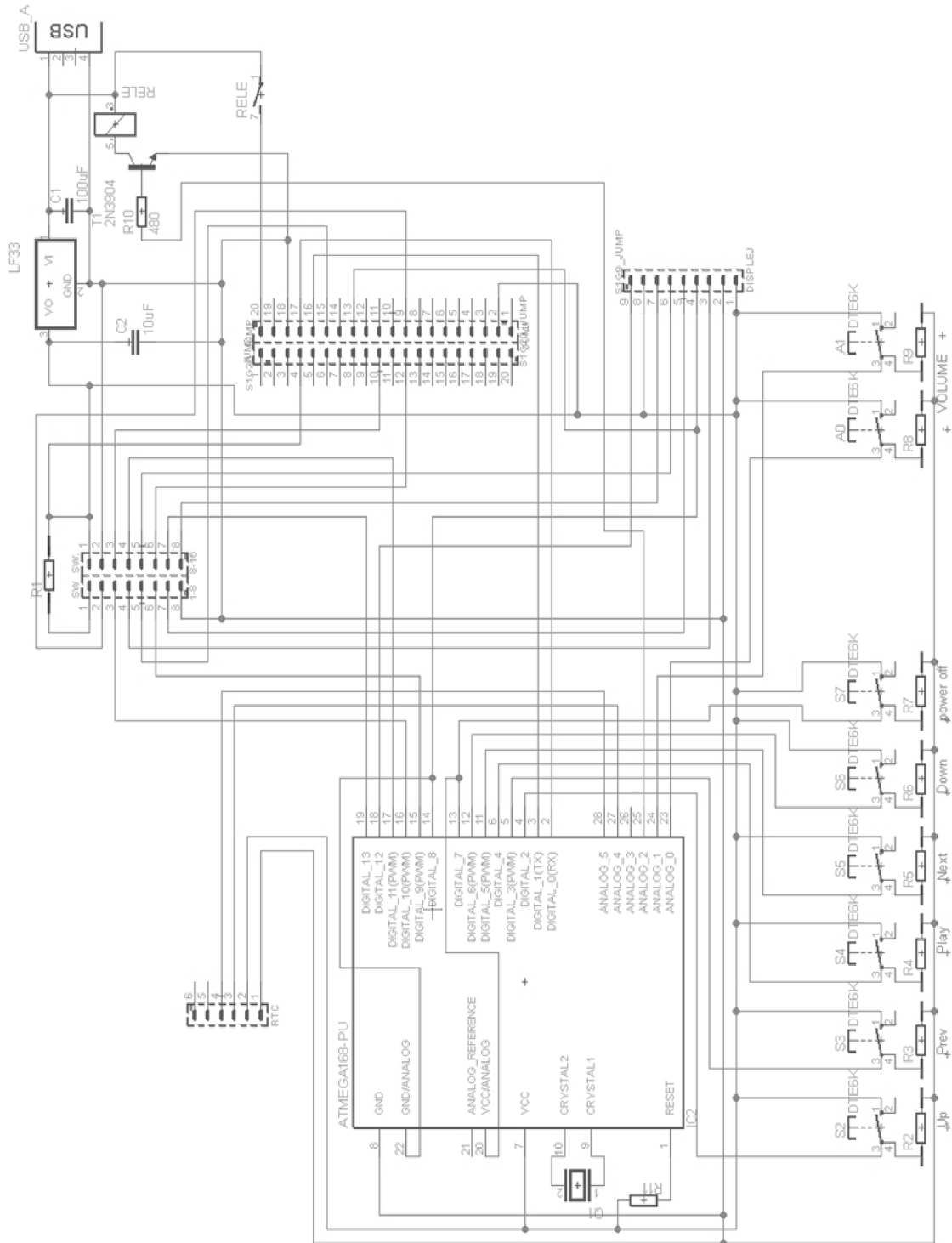
- [20] NXP, Semiconductors. 74CBTLV3257 Quad 1-of-2 multiplexer/demultiplexer: Product data sheet. [online]. s. 19 [cit. 2015-04-12]. Dostupné z: [http://www.nxp.com/documents/data\\_sheet/74CBTLV3257.pdf](http://www.nxp.com/documents/data_sheet/74CBTLV3257.pdf)
- [21] RelaXXPlayer. [online]. [cit. 2015-04-19]. Dostupné z: <http://relaxx.dirk-hoeschen.de/>
- [22] Fritzing [online]. [cit. 2015-05-14]. Dostupné z: <http://fritzing.org/home/>
- [23] Raspberry Pi GEEK [online]. [cit. 2015-05-14]. Dostupné z: <http://www.raspberry-pi-geek.com/>
- [24] COSMO - Reed Relay S1A05 [online]. [cit. 2015-05-20]. Dostupné z: <http://pdf1.alldatasheet.com/datasheet-pdf/view/139543/COSMO/SS1A050000.html>
- [25] Tranzistor 2N3904: datasheet [online]. [cit. 2015-05-22]. Dostupné z: <http://www.gme.cz/img/cache/doc/215/003/bipolarni-tranzistor-2n3904-datasheet-3.pdf>
- [26] Regulátor LF33CV: datasheet [online]. [cit. 2015-05-22]. Dostupné z: <http://www.gme.cz/img/cache/doc/330/165/lf33cv-datasheet-1.pdf>
- [27] RTC - Blokové schéma [online]. [cit. 2015-06-01]. Dostupné z: <https://lh4.googleusercontent.com/8a0LCDGkVas/TYX9YpVU8MI/AAAAAAAAABuU/qeTihRXJExI/s1600/DS3132.png>
- [28] RTC [online]. [cit. 2015-06-01]. Dostupné z: <http://www.maximintegrated.com/en/products/digital/real-time-clocks/DS3231.html>
- [29] RTC datasheet [online]. [cit. 2015-06-01]. Dostupné z: <http://www.maximintegrated.com/en/products/digital/real-time-clocks/DS3231.html#popuppdf>
- [30] Atmega datasheet [online]. [cit. 2015-06-01]. Dostupné z: [http://www.mouser.com/pdfdocs/Gravitech\\_ATMEGA328\\_datasheet.pdf](http://www.mouser.com/pdfdocs/Gravitech_ATMEGA328_datasheet.pdf)

## 9 ZOZNAM PRÍLOH

A	Multimediálny panel .....	43
A.1	Schéma zapojenia.....	43
A.2	Doska plošných spojov .....	44
A.3	Zoznam súčiastok.....	45
B	Obsah CD s prílohou .....	46

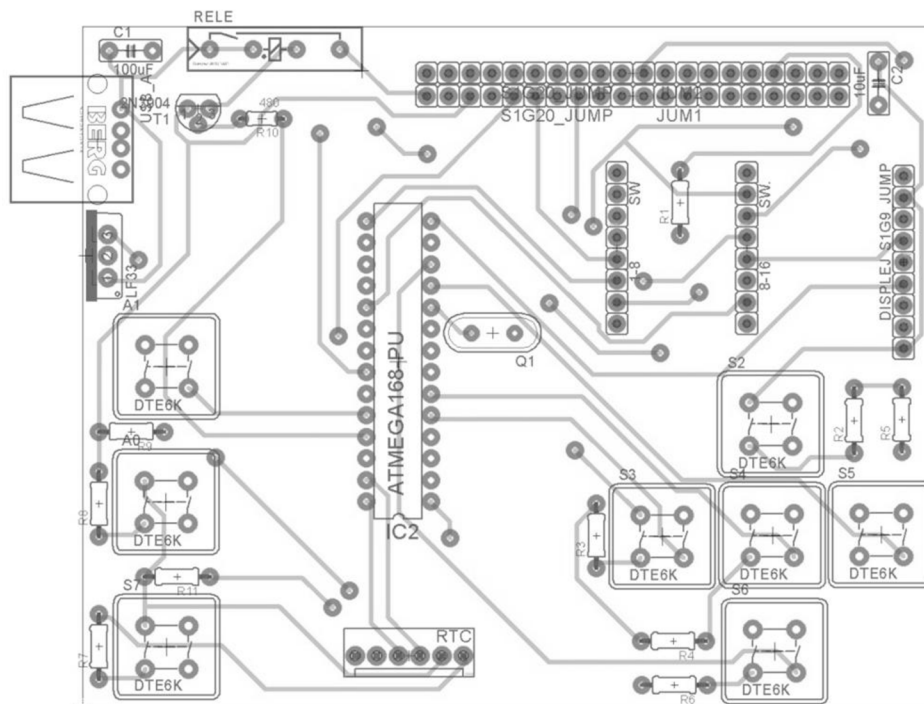
# A Multimediálny panel

## A.1 Schéma zapojenia

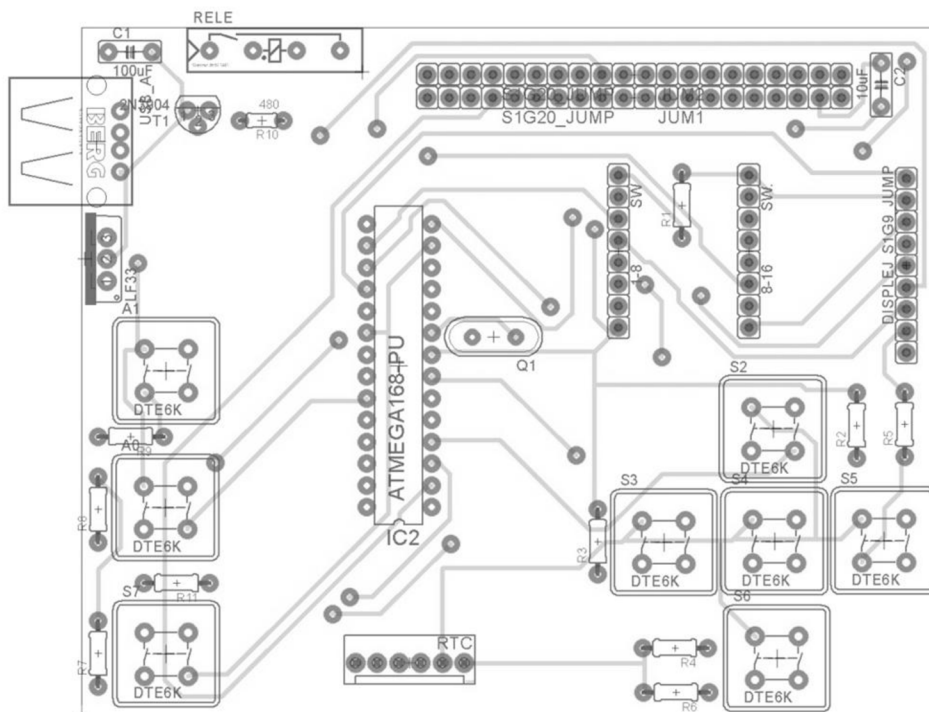


Obr.19: Schéma zapojenia multimediálneho panela

## A.2 Doska plošných spojov



Obr.21: Návrh dosky plošného spoja aj s osadením súčiastok – horná strana



Obr.20: dosky plošného spoja aj s osadením súčiastok – dolná strana

### A.3 Zoznam súčiastok

Označenie	Hodnota	Puzdro	Popis
A0	DTE6K	DTE6K	Tlačidlo
A1	DTE6K	DTE6K	Tlačidlo
C1	100uF	C5B2.5	Kondenzátor
C2	10uF	C5B2.5	Kondenzátor
DISPLEJ	S1G9_JUMP	S1G9_JUM	Radove konektory - koliky - 9x
IC2	ATMEGA168-PU	DIL28-3	Mikrokontrolér
JUM1	S1G20_JUMP	S1G20_JU	Radove konektory - koliky - 20x
JUM2	S1G20_JUMP	S1G20_JU	Radove konektory - koliky - 20x
LF33		TO220V	Regulátor napätia (5V -> 3,3V)
R1	100k $\Omega$	0204/7	Rezistor
R2	100k $\Omega$	0204/7	Rezistor
R3	100k $\Omega$	0204/7	Rezistor
R4	100k $\Omega$	0204/7	Rezistor
R5	100k $\Omega$	0204/7	Rezistor
R6	100k $\Omega$	0204/7	Rezistor
R7	100k $\Omega$	0204/7	Rezistor
R8	100k $\Omega$	0204/7	Rezistor
R9	100k $\Omega$	0204/7	Rezistor
R10	560 $\Omega$	0204/5	Rezistor
R11	100k $\Omega$	0204/7	Rezistor
RELE		REED	Relé
RTC		PSH02-06P	6 pinový konektor
S2		DTE6K	Tlačidlo
S3		DTE6K	Tlačidlo
S4		DTE6K	Tlačidlo
S5		DTE6K	Tlačidlo
S6		DTE6K	Tlačidlo
S7		DTE6K	Tlačidlo
SW		S1G8_JUM	Radový konektor 8
SW.		S1G8_JUM	Radový konektor 8
T1	2N3904	TO92	NPN Tranzistor
USB_A		PN87520	BERG USB konektor
			Raspberry Pi B+
Displej			LCD displej 240x320
Q1	16000 Hz		Časovač F.C16.000

## **B Obsah CD s prílohou**

- Zložka Eagle ( Schéma zapojenia, doska plošných spoja)
- Zložka Atmega ( Zdrojový kód pre mikrokontrolér, knižnica pre displej)
- Zložka Rpi ( Skripty pre Raspberry Pi, knižnica pre displej, font písma)